Moving Picture, Audio and Data Coding
by Artificial Intelligence
www.mpai.community

MPAI Technical Specification

MPAI Metaverse Model (MPAI-MMM) – Architecture (MMM-ARC)

V1.2

WARNING

Use of the technologies described in this Technical Specification may infringe patents, copyrights, or intellectual property rights of MPAI Members or non-members.

MPAI and its Members accept no responsibility whatsoever for damages or liability, direct or consequential, which may result from the use of this Technical Specification.

Readers are invited to review **Error! Reference source not found.**Notices and Disclaimers.

Technical Specification
MPAI Metaverse Model (MPAI-MMM) – Architecture (MMM-ARC) V1.2

Contents

1 Foreword

The international, unaffiliated, non-profit ***Moving Picture, Audio, and Data Coding by Artificial Intelligence (MPAI)*** organisation was established in September 2020 in the context of:

1. **Increasing** use of Artificial Intelligence (AI) technologies applied to a broad range of domains affecting millions of people
2. **Marginal** reliance on standards in the development of those AI applications
3. **Unprecedented** impact exerted by standards on the digital media industry affecting billions of people

believing that AI-based data coding standards will have a similar positive impact on the Information and Communication Technology industry.

The design principles of the MPAI organisation as established by the MPAI Statutes are the development of AI-based Data Coding standards in pursuit of the following policies:

1. Publish upfront clear Intellectual Property Rights licensing frameworks.
2. Adhere to a rigorous standard development process.
3. Be friendly to the AI context but, to the extent possible, remain agnostic to the technology thus allowing developers freedom in the selection of the more appropriate – AI or Data Processing – technologies for their needs.
4. Be attractive to different industries, end users, and regulators.
5. Address five standardisation areas:
    1. *Data Type*, a particular type of Data, e.g., Audio, Visual, Object, Scenes, and Descriptors with as clear semantics as possible.
    2. *Qualifier*, specialised Metadata conveying information on Sub-Types, Formats, and Attributes of a Data Type.
    3. *AI Module* (AIM), processing elements with identified functions and input/output Data Types.
    4. *AI Workflow* (AIW), MPAI-specified configurations of AIMs with identified functions and input/output Data Types.
    5. *AI Framework* (AIF), an environment enabling dynamic configuration, initialisation, execution, and control of AIWs.
6. Provide appropriate Governance of the ecosystem created by MPAI Technical Specifications enabling users to:
    1. *Operate* Reference Software Implementations of MPAI Technical Specifications provided together with Reference Software Specifications
    2. *Test* the conformance of an implementation with a Technical Specification using the Conformance Testing Specification.
    3. *Assess* the performance of an implementation of a Technical Specification using the Performance Assessment Specification.

4. *Obtain* conforming implementations possibly with a performance assessment report from a trusted source through the MPAI Store.

Today, the MPAI organisation rests on four solid pillars:

1. The MPAI Patent Policy specifies the MPAI standard development process and the Framework Licence development guidelines.
2. *Technical Specification: Artificial Intelligence Framework (MPAI-AIF)* specifies an environment enabling initialisation, dynamic configuration, and control of AIWs in the standard AI Framework environment depicted in Figure 1. An AI Framework can execute AI applications called AI Workflows (AIW). An AIW includes interconnected AI Modules (AIM). MPAI-AIF supports small- and large-scale high-performance components and promotes solutions with improved explainability.

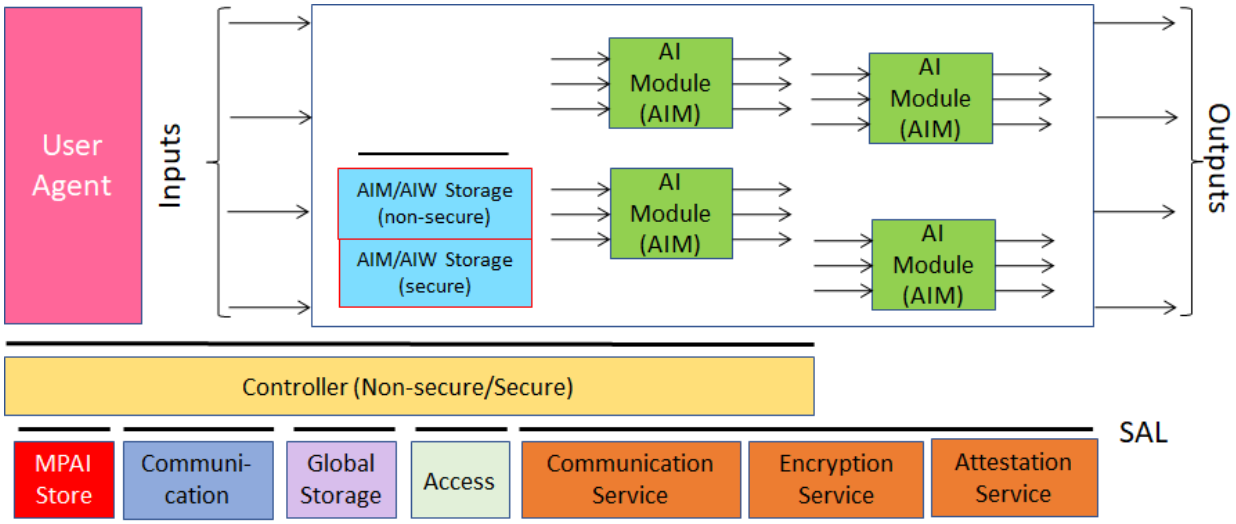

Figure 1 – The AI Framework (MPAI-AIF) V2 Reference Model

3. ***Technical Specification: Data Types, Formats, and Attributes (MPAI-TFA) V1.0*** specifies Qualifiers, a type of metadata supporting the operation of AIMs receiving data from other AIMs. Qualifiers convey information on Sub-Types (e.g., the type of colour), Formats (e.g., the type of compression and transport), and Attributes (e.g., semantic information in the Content). Although Qualifiers are human-readable, they are only intended to be used by AIMs. Therefore, Text, Speech, Audio, and Visual Data exchanged by AIWs and AIMs should be interpreted as being composed of Content (Text, Speech, Audio, and Visual as appropriate) and associated Qualifiers. The specifications of most MPAI Data Types reflect this point.
4. ***Technical Specification: Governance of the MPAI Ecosystem (MPAI-GME) V1.1*** defines the following elements:
    1. Standards, i.e., the ensemble of Technical Specifications, Reference Software, Conformance Testing, and Performance Assessment.
    2. Developers of MPAI-specified AIMs and Integrators of MPAI-specified AIWS (Implementers).
    3. MPAI Store in charge of making AIMs and AIWs submitted by Implementers available to Integrators and End Users.
    4. Performance Assessors, independent entities assessing the performance of implementations in terms of Reliability, Replicability, Robustness, and Fairness.
    5. End Users.

The interaction between and among actors of the MPAI Ecosystem are depicted in Figure 2.

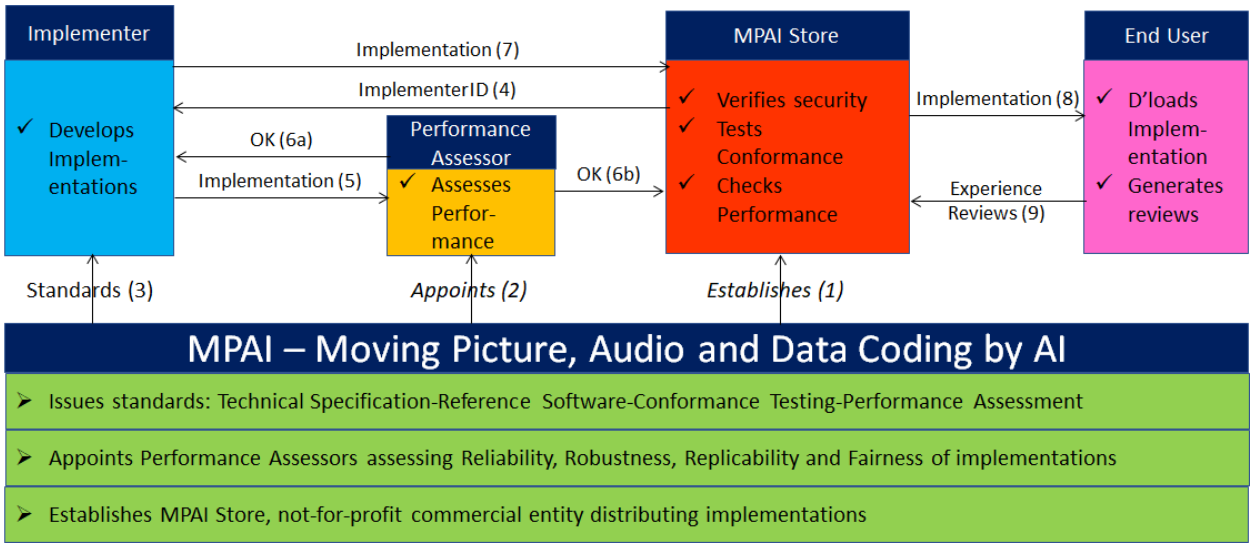

Figure 2 – The MPAI Ecosystem

2 Introduction (informative)

Metaverse is a loose concept considered by many as one of the most promising evolutionary steps of Information and Communication Technology and there are many implementations that can be classified as metaverse instances. So far, however, the metaverse developers made technology decisions that best responded to their needs, often without considering the choices that other developers might have made for similar purposes.

As there have been mounting concerns that such metaverse "walled gardens" do not fully exploit the opportunities offered by current and expected technologies and calls have been made to make metaverse instances "Interoperable", MPAI has developed two Technical Reports and two Technical Specification that provide solutions to the M-Instance Interoperability issues. They are:

1. **Technical Report: MPAI Metaverse Model (MPAI-MMM) – Functionalities** introduces definitions, assumptions for the work, a collection of high-level use cases, a collection of exemplary service providers, a set of ~150 Functionalities, review of the main metaverse-enabling technologies, an analysis of metaverse governance needs, and a standardisation roadmap.
2. **Technical Report: MPAI Metaverse Model (MPAI-MMM) – Functionality Profiles** introduces a revised and extended list of definitions; an operation model of the metaverse based on the notion of Processes performing or requesting other Processes to perform Actions on Items (Items are Data, Metadata, and Qualifiers supported by an M-Instance); an initial identification of Actions, Items, and Basic Data with Use Cases and Functionality Profiles; a collection of representative use cases tested against the Operation Model; and dour initial Functionality Profiles.
3. **Technical Specification – MPAI Metaverse Model (MPAI-MMM) – Architecture (MMM-ARC) V1.2** provides means to achieve M-Instance Interoperability by specifying the Functional Requirements of Processes and Actions. These allow Interoperation of two or more M-Instances that execute Processes, and producing Data that comply with the MMM-ARC Functional Requirements, if necessary via a Conversion Service.
4. **Technical Specification – MPAI Metaverse Model (MPAI-MMM) – Technologies (MMM-TEC) V1.0** specifies or references Items including Qualifiers to enable interoperability between M-Instances supporting the technologies referenced in the Qualifiers.

M-Instance indicates the type of metaverse specified by the two integrated specifications MMM-ARC V1.2 and MMM-TEC V1.0. The Table of Contents of MMM-ARC merges the references to two specifications into one.
In all Chapters and Sections, Terms beginning with a capital letter are defined in *Table 1* if they are specific to this Technical Specification and in *Table 2* if they are common to all MPAI Technical Specifications. All Chapters, Sections, and Annexes are Normative unless they are labelled as Informative.

3 Scope

Technical Specification: MPAI Metaverse Model (MPAI-MMM) – Architecture (MMM-ARC) V1.2 – in the following also called MMM-ARC V1.2 or MMM-ARC – specifies Functional Requirements of Processes and Actions performed by Processes of an M-Instance composed of a set of Processes performing Actions on Items.
In combination with ***Technical Specification: MPAI Metaverse Model (MPAI-MMM) – Technologies (MPAI-TEC) V1.0***, MPAI-ARC V1.2 enables metaverse instances (M-Instances) and Clients to Interoperate, within the constraints set by Profiles, i.e., a client or an M-Instance implemented according to MPAI-ARC V1.2 and MPAI-TEC V1.0 have all the information required to

1. Interpret the Data received from an M-Instance.
2. Act on the Data as intended by the sending M-Instance.
3. Respond to the sending M-Instance with Data that the M-Instance can interpret, act on, and respond to.

The combined MPAI-ARC V1.2 and MPAI-TEC V1.0 contents is:

#	**Chapter name**	**Specified by**	**Status**
1.	**Scope**	MPAI-ARC V1.2	Normative
2.	**Terms**	Shared	Normative
3.	**Operation Model**	Shared	Normative
4.	Functional Requirements of:		
4.1.	**Processes**, i.e., Programs executing in an M-Instance.	MPAI-ARC V1.2	Normative
4.2.	**Actions**, i.e., Functionalities provided by Processes.	MPAI-ARC V1.2	Normative
5.	**Items**, i.e., the Data Types recognised by the M-Instance	MPAI-TEC V1.0	Normative
6.	**Scripting Language**.	MPAI-TEC V1.2	Normative
7.	**Use Cases** leveraging the Tools, i.e., Actions, Items, and Data Types.	MPAI-TEC V1.0	Informative
6.	**Profiles**, i.e., groups of Tools required to satisfy identified needs.	MPAI-TEC V1.0	Normative

This Technical Specification has been developed by the MPAI Metaverse Model (MMM) group of the Requirements Standing Committee. MPAI may decide to publish extensions or new versions of this Technical Specification, or other Technical Specifications of the MPAI-MMM series.

4 Definitions

Capitalised Terms used in MMM-ARC and MMM-TEC have the meaning defined in *Table 1*. Terms applicable to all MPAI Technical Specifications are defined in *Table 2*. Non-capitalised terms letter have the meaning commonly defined for the context in which they are used or represent an entity in the real world. For instance,

1. Table 1 defines *Object*, *Scene*, and *User* but does not define *object*, *scene*, and *human*.
2. Object indicates an Item but object indicates an entity in the Universe commonly classified as object.

A dash "-" preceding a Term in Table 1 means the following:

1. If the font is normal, the Term in the table without a dash and preceding the one with a dash should be placed before that Term. The notation is used to concentrate in one place all the Terms that are composed of, e.g., the word Decentralised followed by one of the words Application, Autonomous Organisation, Finance, System, and User Identifier, or definitions belonging to the same class, e.g., Action and Items.
2. If the font is *italic*, the Term in the table without a dash and preceding the one with a dash should be placed after that Term. The notation is used to concentrate in one place all the Terms that are composed of, e.g., the word Interface preceded by one of the words Brain-Computer, Haptic, Speech, and Visual.
3. If the term is underlined, it is a definition of the components of Actions, Items and Processes.

Table 1 – General Terms and Definitions

Terms	**Definitions**
Account	An Item that uniquely references a human Registered on the M-Instance managing the Account.
Action	A Functionality provided by a Process.
– Authenticate	The Action of requesting that a Service confirm that an Item is what it claims to be.
– Author	The Action of Calling a Service to obtain an Item with associated OutRights to Act on the Item.
– Change	The Action of requesting that a Service modify the Rights of a User and provide OutRights, e.g., to further Change the Rights.
– Convert	The Action of requesting that a Service Modify an Item according to a provided Data Qualifier.
– Discover	The Action of requesting that a Service provide information about Items or Processes satisfying the conditions expressed in the request.
– Execute	The Action of requesting that a Process execute the conditions of a non-executable Contract.
– Hide	The Action of requesting that a Service make the ID of an Item unavailable and provide OutRights, e.g., to make the ID available again.
– Identify	The Action of requesting that a Service produce an Item from Data & Metadata.
– Inform	The Action of requesting that a Service provide information about an Item or Process, such as the Metadata of an Item.
– Interpret	The Action of requesting that a Service provide interpretations of an InItem, such as translation or extraction of Personal Status.
– MM-Add	The Action of requesting that a Service add an Item at an M-Location with a Spatial Attitude and provide OutRights to Act on the MM-Added Item.

– MM-Animate	The Action of requesting that a Service change the features of a Model MM-Embedded at an M-Location with a Stream and provide the OutRights to Act on the MM-Animated Item.
– MM-Disable	The Action of requesting that a Service stop MM-Enabling selected Items Embedded at an M-Location and provide OutRights to Act on the MM-Disabled Entities.
– MM-Embed	The Composite Action of requesting that a Service MM-Add and MM-Enable an Item either located at a Service or at an M-Location at a destination M-Location with a Spatial Attitude and provide OutRights to Act on the MM-Embedded Item.
– MM-Enable	The Action of requesting that a Service accept requests to MM-Send selected Entities MM-Added at an M-Location or to MM-Embed those selected Entities at a destination M-Location and provide OutRights to act on the M-Entities.
– MM-Send	The Action of requesting that a Process forward an Item or Data/Metadata to a Process with appropriate Rights to act on Item or Data/Metadata.
– Modify	The Action of requesting that a Service produce a new Item from an existing Item by providing new Data and Metadata with the OutRights to further Act on the new Item.
– MU-Actuate	The Action of requesting that a Device present an Item available at a Device to a U-Location as Media with a Spatial Attitude.
– MU-Embed	The Composite Action of requesting that: 1. A Service MM-Send selected Entities Embedded at an M-Location to a Device. 2. The Device MU-Actuate the Item received at a U-Location with a Spatial Attitude.
– MU-Send	The Action of requesting that a Process transmit an Item to a Device or store an Item at an Address.
– Post	The Action of requesting that a Marketplace include an Asset to its repertory of Assets.
– Register	The Action of requesting that a Service grant selected Processes of a human the Rights to perform Actions in the M-Instance.
– Resolve	The Action of requesting that a Service forward a Request-Action or a Response-Action to a Resolution Service in another M-Instance.
– Track	The Composite Action of requesting that a Service: 1. MM-Embed a Model at an M-Location with a Spatial Attitude. 2. MU-Animate the Model MM-Embedded at an M-Location. 3. MU-Embed specified Entities at the M-Location to a U-Location.
– Transact	The Action of a $User_1$ (“sender”) requesting that a Service: 1. Assign Rights on an Asset to User2 (“receiver”). 2. Cause: 2.1.Wallet1 of User1 to be increased by Value1. 2.2.Wallet2 of User2 to be decreased by Value2. 2.3.Wallet3 of the Service enabling/facilitating the Transaction to be increased by Value3 (optionally).
– UM-Animate	The Composite Action of a User requesting: 1. A Device to1.1.UM-Capture an animation stream extracted from a scene at a U-Location. 1.2.UM-Send the animation stream and Metadata to a User. 2. A Service to Identify the Animation Stream.

	3. A Service to MM-Animate the Model MM-Embedded at the M-Location using the Animation Stream.
– UM-Capture	The Action of requesting that a Device capture Media from a scene at a U-Location.
– UM-Embed	The Composite Action of a User requesting: 1. A Device to:1.1.UM-Capture a scene at U-Location. 1.2.MM-Send Data and Device-provided Metadata to a User. 2. A Service to: 2.1.Identify an Item from UM-Sent Data and Metadata. 2.2.MM-Embed the Item at an M-Location with a Spatial Attitude.
– UM-Send	The Action of a Device acquiring Data & Metadata from an Address.
– Validate	The Action of requesting that a Service verify that a Process has the Rights to perform or request a Process to perform an Action on an Item.
Avatar	A digital representation of a real or fictitious human.
Blockchain	A shared immutable ledger stored on a peer-to-peer network of computers.
Certification	The attestation that a Process or Item has specified characteristics.
Connected Autonomous Vehicle	(CAV) A vehicle able to autonomously reach a U-Location by using its own sensing and processing capabilities to generate an M-Instance, sharing its M-Instance with other CAVs and issuing actuation commands to its Motion Actuation Subsystem.
Coordinates	A set of numbers used to represent a Position in an M-Instance using a coordinate system.
Conversion	The process of Modifying the Data of an Item according to a provided Data Qualifier.
Data	Information represented in digital form.
– Format	The syntax and semantics of Data.
– Qualifier	Data about Data designed for use by a Machine.
DataMdata	The combination of Data and Metadata that is not (yet) an Item.
Decentralised	
– Application	(dApp) A Process that runs on a decentralised computing system.
– Autonomous Organisation	(DAO) An organisation without centralised leadership, where the main governing rules are typically encoded by means of a Smart Contract.
– Finance	(DeFi) A financial technology based on a secure infrastructure of distributed ledgers like those used by crypto currencies.
– System	A set of dApps enabling a group of Users to make decisions without a centralised entity.
Device	Equipment enabling: – A U-Environment to interact with an M-Instance and/or- An M-Instance to interact with a U-Environment.
Duty	A moral or legal obligation to act or behave.
Entitlement	The state of a User having certain Rights in an M-Instance.
Functional Requirement	A Functionality that is expected to be provided by an entity.
Human	
– *Digital*	Either a Digitised or a Virtual Human.
– *Digitised*	The digital representation of a human.
– *Virtual*	A computer-created Object that has a human appearance when rendered but is not a Digitised Human.

Governance	The action or manner of directing and controlling actors of the Metaverse Ecosystem.
Information and Communication Technologies	(ICT) Technologies that enable the processing and distribution of information via the network.
Interface	A communication pathway enabling systems to interact.
– *Brain-Computer*	(BCI) A communication pathway that allows a human to interact with an M-Instance by sensing and processing the electrical activity of the brain.
– *Haptic*	A communication pathway that allows a human to interact with an M-Instance through bodily movements and sensations.
– *Speech*	A communication pathway that allows a human to interact with an M-Instance using spoken language.
– *Visual*	A communication pathway that allows a human to interact with an M-Instance through bodily movements and visual messages.
Interoperability	The ability of an M-Instance to exchange and make use of the data of another M-Instance as intended by the latter M-Instance.
Item	Data and Metadata supported and identified by an M-Instance.
– Account	An Item that uniquely references a human who has Registered. A User may have more than one Account with one or more Services.
– Activity Data	An Item containing the record of all the Process Actions made by a Process.
– Address	The URL of a storage facility.
– Amount	A number expressing a Value in a Currency.
– Asset	An Item that may be the object of a Transaction. It may be MM-Embedded at an M-Location or Posted to a Service.
– Contract	An Item expressing terms and conditions or a Program that is activated when terms and conditions are met.
– Smart	A Program stored on a Blockchain that runs when activated by an external entity, e.g., a User or another Smart Contract.
– Currency	A medium of exchange enabling Transactions.
– Emotion	The representation of a User's Personal Status that results from its interaction with an environment, such as "Angry", "Sad", "Determined".
– Event	An Item that includes selected Entities at an M-Location and their Animations during a period.
– Identifier	An Item that uniquely references an Item or a Process in an M-Instance.
– M-Environment	An administratively identified subset of an M-Instance.
– Capabilities	(E-Capabilities) An Item expressing the capabilities of an M-Environment.
– Message	An Item containing application-specific Data MM-Sent by a Source Process to a Destination Process.
– M-Instance	An implementation of the MMM-ARC and MMM-TEC Technical Specification.
– M-Location	A region of an M-Instance with Space-Time attributes that is exposed as further subdivided.
– Model	An Object that can be used to spawn other Objects, e.g., by animating them.
– Object	An Item with at least one of Audio, Visual, or Haptic perceptibility attributes.
– *Audio*	An Object perceptible by a hearing device or audible to a human when rendered.
– *Audio-Visual*	An Object whose rendering has both Audio and Visual perceptibility attributes.

– *Visual*	An Object perceptible by a visual device or visible to a human when rendered.
– Orientation	An Item representing an Object's orientation, velocity, and acceleration.
– Persona	A Model representing a human.
– Personal Data	An Item containing a human's Personal Profile, Activity Data of their Users, and Personae.
– Personal Profile	An Item containing a human's Personal Data submitted when Registering with an M-Instance.
– Personal Status	An Item representing the information internal to a User that characterises their behaviour.
– Point of View	The Spatial Attitude of a Persona watching an Environment.
– Position	The Coordinates of a point in a Metaverse Environment using a Coordinate system.
– Program	An Item containing executable code, e.g., application program.
– Provenance	A Data Type containing the list of all Transactions executed on an Asset, first and last included.
– Request-Action	An Item of the request to a Process to perform an Action.
– Response-Action	An Item containing the response of a Process to a Request-Action.
– Rights	An Item expressing the authorisation of a Process to perform Actions on Items at M-Locations during a Time and the Rights Level.
– Level	A type of Right, currently Internal (granted at Registration Time), Acquired (by Process Activity), Granted (by another Process).
– Rules	An Item expressing the terms and conditions under which a Process can perform Action in an M-Instance or M-Environment.
– Scene	A hierarchical Composition of Objects and Scenes having Spatial Attitudes.
– *Basic*	A hierarchical Composition of Objects having Spatial Attitudes.
– Social Attitude	A Personal Status Factor representing the internal state of an Entity related to the way it intends to position itself vis-à-vis the Context, e.g., "Respectful", "Confrontational", "Soothing"..
– Space-Time	An Item representing the combined digital representation of space and time.
– Spatial Attitude	An Item representing the Position and Orientation of an Object, and their velocities and accelerations.
– Stream	An Item made by a continuous temporal flow of Data.
– Time	An Item representing the measure of time.
– Transaction	An Item representing: – The Amount, the WalletID and the Rights on an Asset of a User transferring Rights to another User (Sender). – The Amount, the WalletID and the Rights on on the Asset of another User receiving the Rights (Receiver). – Optionally the Amount and the WalletID of the Service Provider facilitating/enabling the Transaction.
– U-Location	An Item representing a region of the Universe with Space-Time attributes.
– Universe-Metaverse Map	An Item containing a structure establishing a correspondence between U-Locations with M-Locations.
– Value	An Amount and the Currency with which the Amount is expressed.
– Wallet	A container of Values.
– *Crypto*	Software or hardware holding the Public and Private Keys of a User to enable them to make Transactions by accessing their Account on a Blockchain.

Ledger	An Item containing a list of Transactions involving Assets.
Media	Data that is: 1. acquired by a Device sensor whose rendering can be perceived by a human. 2. Can be presented and perceived by a human.
Metadata	Data about Data, e.g., of a human, a Process, or an Item.
Metaverse	
– Actuator	A component of a Device able to MU-Embed an Item to a U-Environment.
– Ecosystem	The ensemble of entities and rules ensuring that Metaverse Instances operate in the interest of Metaverse Stakeholders.
– Enabling Service Layer	The set of Services such as payment, security, identity, privacy, etc. that enable operation of an M-Instance.
– Experience Layer	The set of functions, such as Devices, that generate Experiences.
– Industry	The collection of players that support the design, development, deployment, operation, and content and service provisioning to Metaverse Instances.
– Instance	(M-Instance) A set of Processes providing some or all the following functions: 1. To sense data from U-Locations.2. To process the sensed data and produce Data. 3. To produce one or more M-Environments populated by Objects that can be either digitised or virtual, the latter with or without autonomy. 4. To process Objects from the M-Instance or potentially from other M-Instances to affect U- and/or M-Environments using Objects in ways that are: 4.1.Consistent with the goals set for the M-Instance. 4.2.Effected within the capabilities of the M-Instance. 4.3.Complying with the Rules set for the M-Instance and applicable laws.
– Interoperability	The ability of M-Instance_A to use data from and as intended by M-Instance_B. Interoperability can be Direct or Mediated by a Conversion Service.
– Infrastructure Layer	The set of functions such as network, transport, storage, and (cloud, edge) processing that enable an M-Instance to operate.
– Manager	The entity overseeing the operation of an M-Instance.
– Operation Model	The components and sequence of steps involved in an M-Instance providing Functionalities.
– Operator	The entity overseeing the operation of an M-Environment.
– Partner	A User participating in activities of a Metaverse Operator (i.e., a business customer of an Operator)
– Platform Layer	The set of Services, such as content creation, content discovery, and content access functions that enable an M-Instance to operate.
– Process	The instance of a program being executed.
– Profile	A recognised subset of Technologies specified by MMM-ARC and MMM-TEC.
– Stakeholder	An entity performing a function aimed at achieving a goal in an M-Instance.
– Tool	A Technology or group of Technologies enabling an M-Instance to provide a Functionality.
– Technology	A structured application of scientific and/or technical methods that supports a Functionality.
Object	

– *Audio*	The digital representation of an object or a computer-generated Object that can be rendered to and perceived by a human ear.
– *Autonomous*	A Virtual Object animated by a Process giving it the ability to act (e.g., move, speak, respond, execute) with a degree of autonomy.
– *Composite*	An Object that includes more than one Object Type.
– *Digital*	A Digitised or a Virtual Object.
– *Digitised*	The digital representation of an object.
– *Haptic*	An Object with the haptic features of an object able to be rendered to provide haptic sensations in a human.
– *Human*	An Object representing a human.
– *Speech*	The digital representation of a sound emitted by the vocal tract of a human or generated by a computer with similar audio characteristics.
– Type	One of Audio, Visual, Haptic, Olfaction, and Gustation.
– *Virtual*	A computer-generated Object that is not a Digitised Object.
– *Visual*	The digital representation of an object captured by an electromagnetic or high-frequency audio signal or computer-generated that can be rendered to and perceived by a human eye.
Oracle	A Process providing information from a U-Environment to a Blockchain.
Privacy	The Rights of a User to keep their Personal Profile secret.
Process	An instance of a Program running in a Device or in the computing platform underpinning an M-Instance.
– App	An application-specific Program executed on a Device.
– Capabilities	(P-Capabilities) An Items containing a description of the capabilities of a Process.
– Device	A Process able to: 1. UM-Capture Data from a U-Location2. UM-Send Data and Metadata to a User. and/or 1. MM-Send an Item from an M-Location to the Device. 2. MU-Embed an Item at a U-Location.
– Actuator	The component of a Device able to convert Data into information for the Universe
..- Sensor	The component of a Device able to capture information from the Universe and convert it into Data and Metadata.
– Service	A Process that can be called to provide specific Functionalities.
– *Authoring*	A Service enabling the creation of Items.
– *Conversion*	A Service converting the Data produced by an M-Instance$_A$ into Data understood and acted upon by M-Instance$_B$ as intended by M-Instance$_A$.
– User	A Process representing a Registered human.
Profile	A set of base standards and/or their subsets.
– *Functional*	The set of Functionalities offered by a Metaverse Profile.
– Level	A subdivision of a Profile indicating the completeness of the user experience provided by the Profile.
– *Technology*	The set of Technologies offered by a Metaverse Profile.
Registration	The process whereby a human provides a subset of Personal Data to an M-Instance/Environment to obtain an Account and be authorised to deploy their Processes and Personae.
Rendering	The process of making an Item perceptible by human senses.
Representation	Data in an M-Instance representing an entity of a U-Environment.

Sense of	
– Agency	The subjective awareness of being able to decide, execute, and control one's own actions in an M-Environment.
– Embodiment	The engagement of senses to form a complete M-Instance Experience.
– Presence	The feeling of being in an M-Instance with other Digital Humans for real.
Social Graph	A representation of a User's network of connections with Items, M-Locations, and Processes.
Token	
– *Fungible*	A representation of an Asset that is interchangeable with other Assets of the same type.
– *Non-Fungible*	(NFT) A unique digital identifier of an Asset that: – Cannot be copied (i.e., a copy is known to be a copy), substituted, or subdivided.- Is recorded in a digital ledger. – Is used to certify Object authenticity and ownership.
Trust-less system	A system allowing a User to make reliable Transactions without trusting or knowing the parties the User makes Transactions with.
Universe	The physical world.
– Location	(U-Location) A region of the Universe with Space-Time attributes.
Use Case	An example of how an application domain can be supported by an MMM-ARC and MMM-TEC.
User Keys	The pair of public and private keys where the public key is used to encrypt, and the private key is used to both encrypt and decrypt Data.
User Identifier	
– *Decentralised*	An Identifier that enables the verifiable association with a human without requiring a centralised registry.
– *Self-Sovereign*	A Decentralised Identifier derived from the human's Public Key owned and managed directly by the human based on the knowledge of their own Private Key, e.g., stored in the Crypto Wallet enabled by the Blockchain underpinning the M-Instance.

Table 1 is shared between MMM-ARC V1.2 and MMM-TEC V1.0.
Table 2 includes Terms generally used across MPAI Technical Specifications.

Table 2 – MPAI-wide Terms

Term	Definition
Access	Static or slowly changing data that are required by an application such as domain knowledge data, data models, etc.
AI Framework (AIF)	The environment where AIWs are executed.
AI Model (AIM)	A data processing element receiving AIM-specific Inputs and producing AIM-specific Outputs according to according to its Function. An AIM may be an aggregation of AIMs.
AI Workflow (AIW)	A structured aggregation of AIMs implementing a Use Case receiving AIW-specific inputs and producing AIW-specific outputs according to the AIW Function.
Application Standard	An MPAI Standard designed to enable a particular application domain.
Channel	A connection between an output port of an AIM and an input port of an AIM. The term "connection" is also used as synonymous.
Communication	The infrastructure that implements message passing between AIMs.

Component	One of the 7 AIF elements: Access, Communication, Controller, Internal Storage, Global Storage, Store, and User Agent
Composite AIM	An AIM aggregating more than one AIM.
Component	One of the 7 AIF elements: Access, Communication, Controller, Internal Storage, Global Storage, Store, and User Agent
Conformance	The attribute of an Implementation of being a correct technical Implementation of a Technical Specification.
– Testing	The normative document specifying the Means to Test the Conformance of an Implementation.
– Testing Means	Procedures, tools, data sets and/or data set characteristics to Test the Conformance of an Implementation.
Connection	A channel connecting an output port of an AIM and an input port of an AIM.
Controller	A Component that manages and controls the AIMs in the AIF, so that they execute in the correct order and at the time when they are needed
Data	Information in digital form.
– Format	The standard digital representation of Data.
– Type	An instance of Data with a specific Data Format.
– Semantics	The meaning of Data.
Descriptor	Coded representation of a text, audio, speech, or visual feature.
Digital Representation	Data corresponding to and representing a physical entity.
Ecosystem	The ensemble of actors making it possible for a User to execute an application composed of an AIF, one or more AIWs, each with one or more AIMs potentially sourced from independent implementers.
Explainability	The ability to trace the output of an Implementation back to the inputs that have produced it.
Fairness	The attribute of an Implementation whose extent of applicability can be assessed by making the training set and/or network open to testing for bias and unanticipated results.
Function	The operations effected by an AIW or an AIM on input data.
Global Storage	A Component to store data shared by AIMs.
AIM/AIW Storage	A Component to store data of the individual AIMs.
Identifier	A name that uniquely identifies an Implementation.
Implementation	1. An embodiment of the MPAI-AIF Technical Specification, or 2. An AIW or AIM of a particular Level (1-2-3) conforming with a Use Case of an MPAI Application Standard.
Implementer	A legal entity implementing MPAI Technical Specifications.
ImplementerID (IID)	A unique name assigned by the ImplementerID Registration Authority to an Implementer.
ImplementerID Registration Authority (IIDRA)	The entity appointed by MPAI to assign ImplementerID's to Implementers.

Instance ID	Instance of a class of Objects and the Group of Objects the Instance belongs to.
Interoperability	The ability to functionally replace an AIM with another AIW having the same Interoperability Level
– Level	The attribute of an AIW and its AIMs to be executable in an AIF Implementation and to: 1. Be proprietary (Level 1) 2. Pass the Conformance Testing (Level 2) of an Application Standard 3. Pass the Performance Testing (Level 3) of an Application Standard.
Knowledge Base	Structured and/or unstructured information made accessible to AIMs via MPAI-specified interfaces
Message	A sequence of Records transported by Communication through Channels.
Normativity	The set of attributes of a technology or a set of technologies specified by the applicable parts of an MPAI standard.
Performance	The attribute of an Implementation of being Reliable, Robust, Fair and Replicable.
– Assessment	The normative document specifying the Means to Assess the Grade of Performance of an Implementation.
– Assessment Means	Procedures, tools, data sets and/or data set characteristics to Assess the Performance of an Implementation.
– Assessor	An entity Assessing the Performance of an Implementation.
Profile	A particular subset of the technologies used in MPAI-AIF or an AIW of an Application Standard and, where applicable, the classes, other subsets, options and parameters relevant to that subset.
Record	A data structure with a specified structure
Reference Model	The AIMs and theirs Connections in an AIW.
Reference Software	A technically correct software implementation of a Technical Specification containing source code, or source and compiled code.
Reliability	The attribute of an Implementation that performs as specified by the Application Standard, profile, and version the Implementation refers to, e.g., within the application scope, stated limitations, and for the period of time specified by the Implementer.
Replicability	The attribute of an Implementation whose Performance, as Assessed by a Performance Assessor, can be replicated, within an agreed level, by another Performance Assessor.
Robustness	The attribute of an Implementation that copes with data outside of the stated application scope with an estimated degree of confidence.
Scope	The domain of applicability of an MPAI Application Standard
Service Provider	An entrepreneur who offers an Implementation as a service (e.g., a recommendation service) to Users.

Standard	A set of Technical Specification, Reference Software, Conformance Testing, Performance Assessment, and Technical Report of an MPAI application Standard.
Technical Specification	(Framework) the normative specification of the AIF. (Application) the normative specification of the set of AIWs belonging to an application domain along with the AIMs required to Implement the AIWs that includes: 1. The formats of the Input/Output data of the AIWs implementing the AIWs. 2. The Connections of the AIMs of the AIW. 3. The formats of the Input/Output data of the AIMs belonging to the AIW.
Testing Laboratory	A laboratory accredited to Assess the Grade of Performance of Implementations.
Time Base	The protocol specifying how Components can access timing information
Topology	The set of AIM Connections of an AIW.
Use Case	A particular instance of the Application domain target of an Application Standard.
User	A user of an Implementation.
User Agent	The Component interfacing the user with an AIF through the Controller
Version	A revision or extension of a Standard or of one of its elements.
Zero Trust	A cybersecurity model primarily focused on data and service protection that assumes no implicit trust.

5 References

5.1 Normative reference

1. MPAI; Technical Specification: Context-based Audio Enhancement (MPAI-CAE) V2.2.
2. MPAI; Technical Specification: MPAI Metaverse Model (MPAI-MMM) – Technologies (MMM-TEC) V1.0.
3. MPAI; Technical Specification: Object and Scene Description (MPAI-OSD) V1.1.
4. MPAI; Technical Specification: Portable Avatar Format (MPAI-PAF) V1.2.
5. MPAI; Technical Specifications: AI Module Profiles (MPAI-PRF) V1.0.
6. MPAI; Technical Specification: Data Types, Formats, and Attributes (MPAI-TFA) V1.1.
7. ECMA; ECMA-404 The JSON Data Interchange Standard.

5.2 Informative references

8. MPAI; The MPAI Statutes.
9. MPAI; The MPAI Patent Policy.
10. MPAI; Technical Report – MPAI Metaverse Model (MPAI-MMM) – Functionalities; January 2023.

11. MPAI; Technical Report – MPAI Metaverse Model (MPAI-MMM) – Functionality Profiles; May 2023;
12. MPAI; Technical Specification: Governance of the MPAI Ecosystem (MPAI-GME) V1.1.
13. MPAI; Technical Specification: Artificial Intelligence Framework (MPAI-AIF) V2.0.
14. MPAI; Technical Specification – Connected Autonomous Vehicle (MPAI-CAV) – Architecture V1.1.
15. MPAI; Technical Specification – Connected Autonomous Vehicle (MPAI-CAV) – Technologies V1.0.
16. MPAI; MPAI; Framework Licence: MPAI Metaverse Model (MPAI-MMM) – Architecture.
17. MPAI; MPAI; Framework Licence: MPAI Metaverse Model (MPAI-MMM) – Technologies.

6 Operation (Informative)

MPAI-MMM, in the following also called *MMM*, indicates the combined *MMM-ARC* and *MMM-TEC* Technical Specifications for M-Instance interoperability. It defines a metaverse instance (M-Instance) as a platform offering a subset or all of the following functions:

1. Senses Data from a M-Instance of the *Universe*, i.e., the real world.
2. Transforms the sensed Data into processed Data.
3. Produces one or more M-Environments (i.e., subsets of an M-Instance) populated by *Objects* that can be one of the following:
    1. *Digitised* – sensed from the Universe – possibly animated by activities in the Universe.
    2. *Virtual* – imported or internally generated – possibly autonomous or driven by activities in the Universe.
    3. *Mixed*.
4. Acts on Objects from the M-Instance or potentially from other M-Instances on its initiative, or driven by the actions of humans or machines in the Universe.
5. Affects U- and/or M-Environments using Objects in ways that are:
    1. Consistent with the goals set for the M-Instance.
    2. Within the Capabilities of the M-Instance (M-Capabilities).
    3. According to the Rules of the M-Instance.
    4. Respecting applicable laws and regulations.

The functionalities of an M-Instance are provided by a set of Processes performing Actions on Items (i.e., instances of MMM-specified Data Types that have been Identified in the M-Instance) with various degrees of autonomy and interaction. An implementation may merge some MMM-specified Processes into one or split an MMM-specified Process into more than one process provided that the resulting system behaves as specified by MMM.

Some Processes exercise their activities strictly inside the M-Instance, while others have various degrees of interaction with Data sensed from or actuated in the Universe. Processes may be characterised as:

1. *Services* providing specific functionalities, such as content authoring.
2. *Devices* connecting the Universe to the M-Instance and the M-Instance to the Universe.
3. *Apps* running on Devices.
4. *Users* representing and acting on behalf of human entities residing in the Universe. A User (but other types of Process as well) may be rendered as a Persona, i.e., a static or dynamic avatar.

Processes perform their activities by communicating with other Processes or by performing *Actions* on *Items*. Examples of Items are Asset, 3D Model, Audio Object, Audio-Visual Scene, etc. MMM specifies the *Functional Requirements* of some 30 Actions, i.e., Functionalities that are performed by Processes, and some 60 Items.

A Process holds a list of Process Actions, each of which expresses the Action that the Process has performed or may perform on certain Items at certain M-or U-Locations (areas of the virtual and real world, respectively) during certain Times.
For convenience, prefixes are added to Action names:

1. *MM* indicates Actions performed inside the M-Instance, e.g., MM-Animate is the Action that uses a stream to animate a 3D Model with a Spatial Attitude (defined as Position, Orientation, and their velocities and accelerations).
2. *MU* indicates Actions in the M-Instance influencing the Universe, e.g., MU-Actuate is the Action of a Device rendering one of its Items to a U-Location as *Media* with a Spatial Attitude.
3. *UM* indicates Actions in the Universe influencing the M-Instance, e.g., UM-Embed is the Action of placing an Item produced by *Identify*ing a scene, UM-Capture.*d* at a U-Location, at an M-Location with a Spatial Attitude.

Some Actions, such as UM-Embed, are Composite Actions, i.e., combinations of Basic Actions.
Rights are a basic notion underpinning the operation of the MMM and are defined as the list of combinations of *Process Action* and *Level*. Process Action is an Item including Action, the Items on which it can be performed, the M-Locations or U-Locations where it can be performed and the Times during which it can be performed. Levels indicate that the Rights are *Internal*, i.e., assigned by the M-Instance at Registration time, *Acquired*, i.e., obtained by initiative of the Process, or *Granted* to the Process by another Process. The way an M-Instance verifies the compliance of a Process to its Rights in not specified. An M-Instance can decide to verify the Activity Data (the log of all performed Process Actions) based on claims by another Process, to make random verifications or to not make any verification at all.
A Process can request another Process to perform an Action on its behalf by using the *Inter-Process Protocol*. If the requested Process is in another M-Instance, it will use the *Inter-M-Instance Protocol* to request a *Resolution Service* of its M-Instance to establish a communication with another Resolution Service in the other M-Instance. The Backus Naur form of the *MMM-Script* enables efficient communication between Processes.
Figure 1 gives a summary view of these basic MMM notions.

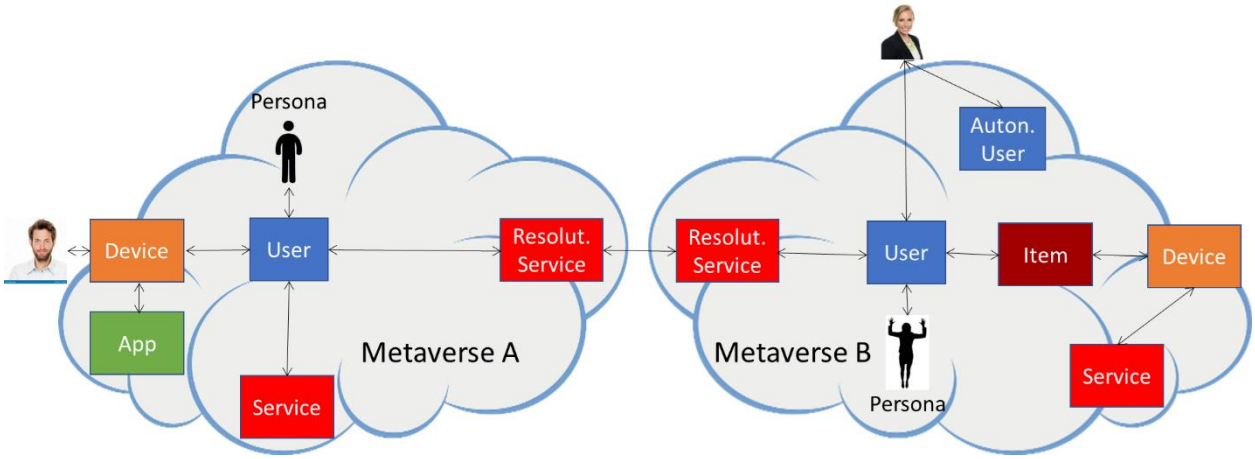

Figure 1 – Inter-Process/M-Instance Communication

To be admitted to an M-Instance, a human may be requested to provide a subset of their Personal Profile and to Transact a Value (i.e., an *Amount* in a Currency). The M-Instance then grants certain Rights to identified Processes of the Registered human, including the import of *Persona*e (i.e., their avatars) for their Users.
The fast development of certain technology areas is one of the issues that has prevented the development of standards for metaverse interoperability. MMM deals with this issue by

providing the JSON syntax and semantics for all Items. When needed, the JSON syntax references *Qualifiers*, MPAI-defined Data Types that provide additional information to the Data Type in the form of:

1. *Sub-Type* (e.g., the colour space of a Visual Data Type).
2. *Format* (e.g., the compression or the file/streaming format of Speech).
3. *Attributes* (e.g., the Binaural Cues of an Audio Object).

An M-Instance or a Client receiving a Visual Object can understand whether it has the required technology to process that Visual Object, or else it should rely on a *Conversion Service* to obtain a version of the Object suitable to the M-Instance or Client.

A M-Instance can be a costly undertaking if all technologies required by the MMM Technical Specification need to be implemented even for M-Instances of a limited scope. MMM-Profiles are introduced to facilitate the take-off of the metaverse. A Profile only includes a subset of Actions and Items that are expected to be needed by a sizeable number of applications. MMM defines four Profiles:

1. Baseline Profile enables a human equipped with a Device supporting the Baseline Profile that enables basic applications such as lecture, meeting, and hang-out.
2. Finance Profile enables a human equipped with a Device supporting the Finance Profile to perform trading activities.
3. Management Profile includes the functionalities of the Baseline and Finance Profiles and enables a controlled ecosystem with more advanced functionalities.
4. High Profile enables all the functionalities of the Management Profile with a few additional functionalities of its own.

MPAI did develop some use cases in the two MPAI-MMM Technical Reports published in 2022. They were used to develop the MMM-ARC and MMM-TEC Technical Specifications. MMM includes several Verification Use Cases that use MMM-Script to verify that the currently specified Actions and Items enable full support of those identified Use Cases.

7 Processes

7.1 Introduction

Process is the first of the constitutive elements of MMM'. It is an instance of a Program running in a device or in the computing platform underpinning an M-Instance specified by:

1. The Functions it performs.
2. Qualifiers.
3. Metadata having the following extensible general form:

ProcessID	The ID of the Process.
InRights	The list of Process Actions the Process can perform with their Levels.
OutRights	The list of Process Actions that another Process can request the Process to perform.
WalletID	The ID of the Wallet related to the Process.
DescrMdata	Any human-readable description of the Process.

4. Performs Actions on Items.
5. May request other Processes to perform Actions (**Inter-Process Protocol**) by sending a Request-Action Item.
6. Performs a request contained in a Request-Action if:
    - The requesting Process holds the Rights that are required to perform the request.
    - The requested Process holds Rights to perform the requested Action on the Item.
7. May Perform, or request another Process to perform, Actions on Items even in the absence of Rights, if the Rules so allow.
8. May send back a Response-Action after receiving a Request-Action.

Table 1 – Elements of Request-Action and Response-Action

Request-Action	**Response-Action**	**Details**
Request-Action ID	Response-Action ID	Unique ID
Emission Time	Emission Time	Tine of Issuance
Source Process ID	Source Process ID	Requesting Process ID
Destination Process ID	Destination Process ID	Requested Process ID
InItems	OutItems	In/Output Items required by the Action
InLocations		Locations of InItems
OutLocations		Locations of OutItems
OutRights		Expected Rights on OutItems

10. May communicate to a Process in another M-Instance through an M-Instance's Resolution Service (**Inter M-Instance Protocol**).
11. To obtain conversion of the Format of an Item's Data by calling a Conversion Service (see Figure 2).
12. To specify their communication needs by:
    - Requesting the needed maximum and average bitrate value.
    - Reserving the needed bitrate for a time and a location.
    - Requesting that the same simultaneous Experience be provided to a specified number of Devices.

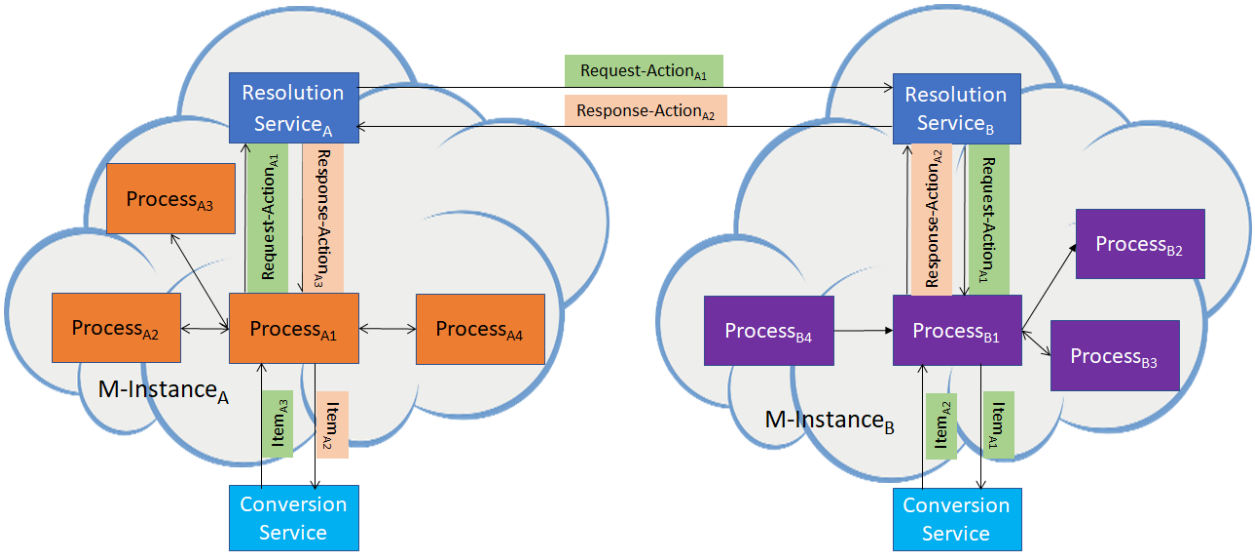

Figure 2 – Resolution and Conversion Services

There are four types of Process.

7.2 App

Functions	An instance of an application-specific Program executed on a Device.
Functional Requirements	The Manager of the M-Instance in which an App will be deployed may request that the Device be subject to certification.

7.3 Device

Functions	A Device can: 1. UM-Capture Data from a U-Location 2. UM-Send Data and Metadata to a User and/or 1. MM-Send an Entity from an M-Location to the Device 2. MU-Render an Entity at a U-Location.
Functional Requirements	To connect and interoperate with an M-Instance, a Device needs to expose its Process Capabilities to the M-Instance.

7.4 Service

Purpose	A Process that provides specific Functionalities.
Functional Requirements	A Service may be: 1. One of the Services natively supported by an M-Instance. 2. Hosted by the M-Instance but provided by a third party.

7.5 User

Purpose	A Process representing a Registered human.
Functional Requirements	1. A User may perform the following functions: 1.1. Act as the interface of the human with the M-Instance. 1.2. Render the User as a Persona UM-Animated by a Stream or MM-Animated by an autonomous agent. 2. Animation results from an MM-/UM-Animate Action and enabled by a Program run by the User. 3. The Animation Program may be part of the Processes registered by a human or provided by the M-Instance.

8 Actions

This chapter specifies purpose, functional requirements, and metadata for the Actions that a Process can perform in an M-Instance. They are grouped in 5 categories

8.1 General Actions

8.1.1 Register

Purpose	The Action of a human requesting that a Service grant their elected Users the Rights to perform Actions in the M-Instance.	
Request-Action	Time	Request is issued
	Source	humanID
	Destination	ServiceID
	Requested Action	*Register*
	InItem	PersonalData ∨ PersonalDataID
	InLocation	Address

	OutLocation	ServiceID
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItem	Account ∨ AccountID
	Error	FaultyReq	The Request is Faulty
		Wallet	Insufficient Value

8.1.2 Change

Purpose	The Action of modifying the Rights of a User and providing OutRights, e.g., to further Change the Rights.	
Request-Action	Time	Request is issued
	Source	UserID
	Destination	ServiceID
	Action	*Change*
	InItems	UserID1 ∧ (Rights ∨ RightsID)
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.1.3 Hide

Purpose	The Action of making the ID of an Item unavailable and providing OutRights, e.g., to make the ID available again.	
Request-Action	Time	Request is issued
	Source	UserID
	Destination	ServiceID
	Requested Action	*Hide*
	InItem	Item ∨ ItemID
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.1.4 Authenticate

Purpose	The Action of requesting to confirm that an Entity is what it claims to be.	
Request-Action	Time	Request is issued
	Source	UserID
	Destination	ServiceID
	Action	*Authenticate*
	InItems	ServiceRequest ∨ ServiceRequestID
	InLocation	M-LocationID ∨ UserID1
	OutLocation	UserID
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItem	ServiceResponse ∨ ServiceResponseID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		M-Location	Out of range
		U-Location	Out of range

8.1.5 Identify

Purpose	The Action of requesting to produce an Item from Data & Metadata.		
Request-Action	Time	Request is issued	
	Source	Process ID	
	Destination	ServiceID	
	Action	*Identify*	
	InItems	DataMdata	
	InLocation	UserID	
	OutLocation	ServiceID	
	OutRight	Rights ∨ RightsID	
Response-Action	Success	OutItem	Item ∨ ItemID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.1.4 Modify

Purpose	The Action of requesting to produce a new Item from an existing Item by providing new Data and Metadata with the OutRights to further Act on the new Item.		
Request-Action	Time	Request is issued	
	Source	Process ID	
	Destination	ServiceID	
	Action	*Modify*	
	InItems	Item ∧ DataMdata	
	InLocation	ServiceID ∨ UserID	
	OutLocation	ServiceID	
	OutRight	Rights ∨ RightsID	
Response-Action	Success	OutItem	Item ∨ ItemID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.1.5 Validate

Purpose	The Action of verifying that a Process has the Rights to perform or request a Process to perform a Process Action.	
Request-Action	Time	Request is issued
	Source	ProcessID
	Destination	ProcessID

	Action	*Validate*	
	InItem	Request-Action ∨ Request-ActionID	
	InLocation	ProcessID	
	OutLocation	ProcessID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Item ∨ ItemID
	Error	Request	Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.1.6 Execute

Purpose	The Action of executing a Contract.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ProcessID	
	Action	*Execute*	
	InItem	Contract ∨ ContractID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.2 Call a Service

8.2.1 Author

Purpose	The Action of requesting an Item with associated OutRights to Act on the Item.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*Author*	
	InItems	Item ∨ ItemID ∨ DataMdata	
	InLocation	UserID ∨ Address	
	OutLocation	UserID ∨ ServiceID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Item ∨ ItemID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Wallet error	Insufficient Value

8.2.2 Discover

Purpose	The Action of requesting a to provide Item IDs or Process ID relevant to the request to Discover.

Request-Action			
	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*Discover*	
	InItem	ServiceRequest ∨ ServiceRequestID	
	InLocation	UserID ∨ ServiceID	
	OutLocation	UserID	
	OutRights	Rights ∨ RightsID	
Response-Action			
	Success	OutItem	ServiceResponse ∨ ServiceResponseID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.2.3 Inform

Purpose	The Action of requesting information about an Item or Process, such as Metadata.		
Request-Action			
	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*Inform*	
	InItem	ServiceRequest ∨ ServiceRequestID	
	InLocation	M-LocationID	
	OutLocation	UserID	
	OutRights	Rights ∨ RightsID	
Response-Action			
	Success	OutItem	ServiceResponse ∨ ServiceResponseID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.2.4 Interpret

Purpose	The Action of requesting interpretation of an Item, such as translation or extraction of Personal Status.		
Request-Action			
	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*Interpret*	
	InItem	ServiceRequest ∨ ServiceRequestID	
	InLocation	M-LocationID ∨ ServiceID	
	OutLocation	UserID	
	OutRights	Rights ∨ RightsID	

Response-Action	Success	OutItem	ServiceResponse ∨ ServiceResponseID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.2.5 Post

Purpose	The Action of requesting that a Marketplace include an Asset to its repertory of Assets.	
Request-Action	Time	Request is issued
	Source	UserID
	Destination	ServiceID
	Action	*Post*
	InItem	Asset ∨ AssetID
	InLocation	UserID ∨ ServiceID
	OutLocation	ServiceID
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Wallet	Insufficient Value

8.2.6 Transact

Purpose	The Action of a $User_1$ (“sender”) requesting that a Service: 1. Assign Rights on an Asset to $User_2$ (“receiver”). 2. Cause: 2.1. $Wallet_1$ of $User_1$ to be increased by $Value_1$. 2.2. $Wallet_2$ of $User_2$ to be decreased by $Value_2$. 2.3. $Wallet_3$ of the Service enabling/facilitating the Transaction to be increased by $Value_3$ (optionally).

Request-Action	Time	Request is issued
	Source	UserID
	Destination	ServiceID
	Action	*Transact*
	InItem	Transaction ∨ TransactionID
	InLocation	UserID ∨ ServiceID
	OutLocation	UserID ∨ ServiceID
	OutRights	Rights ∨ RightsID

Response-Action			
	Success	OutItems	AssetID ∧ $WalletID_1$ ∧ $WalletID_2$ ∧ WalletID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Wallet	$Wallet_2$ has insufficient Value

8.2.7 Convert

Purpose	The Action of requesting changing the Data of an Item according to a given Qualifier.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*Convert*	
	InItem	(Item ∨ ItemID) ∧ FormatID	
	InLocation	ServiceID ∨ M-LocationID	
	OutLocation	ServiceID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Item ∨ ItemID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.2.8 Resolve

Purpose	The Action of requesting that a Service in an M-Instance forward a Request-Resolve or a Response-Resolve Item to a Resolution Service of another M-Instance.		
Request-Action	Time	Request is issued	
	Source	ProcessID	
	Destination	ServiceID	
	Requested Action	*Resolve*	
	InItem	(Request-Action ∨ Request-ActionID) ∨ (Response-Action ∨ Response-ActionID)	
	InLocation	ProcessID	
	OutLocation	ProcessID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Item ∨ ItemID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.3 Manage Entities (Metaverse to Metaverse)

8.3.1 MM-Add

Purpose	The Action of adding an Entity at an M-Location with a Spatial Attitude and providing OutRights to Act on the MM-Added Entity.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*MM-Add*	
	InItem	(Entity ∨ EntityID) ∧ Spatial Attitude	
	InLocation	UserID ∨ ServiceID ∨ M-LocationID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Entity ∨ EntityID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Clash	Entity clashes with another Entity
		M-Location	Out of range

8.3.2 MM-Animate

Purpose	The Action of MM-Embedding a Model at an M-Location using a Process to change its features and providing the OutRights to Act on the MM-Added Model.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ProcessID	
	Action	*MM-Animate*	
	InItem	(Model ∨ ModelID) ∧ Spatial Attitude	
	InLocation	ServiceID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Item mismatch	Entity and Animation Stream Data Types.

8.3.3 MM-Disable

Purpose	The Action of stopping to MM-Enable selected Entities Embedded at an M-Location and providing OutRights to Act on the MM-Disabled Entities.	
Request-Action	Time	Request is issued
	Source	UserID
	Destination	ServiceID
	Action	*MM-Disable*
	InItem	List of EntityIDs
	InLocation	M-LocationID

	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		M-Location	Out of range

8.3.4 MM-Embed

Purpose	The Composite Action of requesting that a Service MM-Add and MM-Enable an Entity either located at a Service or at an M-Location at a destination M-Location with a Spatial Attitude and providing OutRights to Act on the MM-Embedded Entity.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*MM-Embed*	
	InItem	(Entity ∨ EntityID) ∧ Spatial Attitude	
	InLocation	ServiceID ∨ M-LocationID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Clash	Entity clashes with another Entity
		M-Location	Out of range

8.3.5 MM-Enable

Purpose	The Action of implementing requests to MM-Send selected Items MM-Added at an M-Location per the Rights of the requesting User and providing OutRights to act on the selected M-Entities.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*MM-Enable*	
	InItem	Entity ∨ EntityID	
	InLocation	M-LocationID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		M-Location	Out of range

8.3.6 MM-Send

Purpose	The Action of sending an Item, or Data/Metadata to a Process with OutRights given to the Destination Process to Act on the Item or Data/Metadata.		
Request-Action	Time	Request is issued	
	Source	ProcessID	
	Destination	ProcessID	
	Action	*MM-Send*	
	InItem	Item ∨ ItemID ∨ DataMdata	
	InLocation	ProcessID ∨ M-Location	
	OutLocation	ProcessID ∨ M-Location	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Item ∨ ItemID ∨ DataMdata
	Error	Request	Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

8.4 Manage Entities (Metaverse to Universe)

8.4.1 MU-Actuate

Purpose	The Action of presenting an Item available at a Device to a U-Location as Media with a Spatial Attitude.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	DeviceID	
	Action	*MU-Actuate*	
	InItem	(Entity ∨ EntityID) ∧ Spatial Attitude	
	InLocation	DeviceID	
	OutLocation	U-LocationID	
	OutRights	Metadata	
Response-Action	Success	OutItem	Media
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		U-Location	Out of range

8.4.2 MU-Embed

Purpose	The Composite Action of requesting that: 1. A Service MM-Send an Scene MM-Embedded at an M-Location to a Device. 2. The Device MU-Actuate Scene at a U-Location with a Spatial Attitude.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*MU-Embed*	
	InItem	Scene ∧ Spatial Attitude	
	InLocation	M-Location	

	OutLocation	U-Location	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	Action result	Media
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		M-Location	Out of range

8.4.3 MU-Send

Purpose	The Action of requesting that a Process store an Item at an Address.		
Request-Action	Time	Request is issued	
	Source	ProcessID	
	Destination	ServiceID	
	Action	*MU-Send*	
	InItem	Item ∨ ItemID	
	InLocation	M-LocationID ∨ ProcessID	
	OutLocation	Address	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Address	Incorrect

8.4.4 Track

Purpose	The Composite Action of requesting that a Service: 1. MM-Embed a Model at an M-Location with a Spatial Attitude. 2. UM-Animate the Model MM-Embedded at an M-Location. 3. MU-Embed specified Entities at the M-Location to a U-Location.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	ServiceID	
	Action	*Track*	
	InItem	(Model ∨ ModelID) ∧ Spatial Attitude ∧ M-LocationID	
	InLocation	ServiceID	
	OutLocation	U-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Media
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		M-LocationID	Out of range
		U-LocationID	Out of range

8.5 Manage Entities (Universe to Metaverse)

8.5.1 UM-Animate

Purpose	The Composite Action of requesting: 1. A Device to 1. UM-Capture an animation stream extracted from an object at a U-Location. 2. UM-Send the animation stream and Metadata to a User. 2. A Service to Identify the Animation Stream. 3. A Service to MM-Embed a Model at anM-Location and MM-Animate it using the Animation Stream.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	DeviceID	
	Action	*UM-Animate*	
	InItem	object ∧ (Model ∨ ModelID)	
	InLocation	U-LocationID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Entity ∨ EntityID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		U-Location	Out of range
		M-Location	Out of range

8.5.2 UM-Capture

Purpose	The Action of capturing Media from a scene at a U-Location.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	DeviceID	
	Action	*UM-Capture*	
	InItem	scene	
	InLocation	U-LocationID	
	OutLocation	DeviceID	
Response-Action	Success	OutItem	Media
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		U-Location	Out of range

8.5.3 UM-Embed

Purpose	The Composite Action of requesting: 1. A Device to: 1. UM-Capture a scene at U-Location. 2. MM-Send Data and Device-provided Metadata to a User.

	2. A Service to: 1. Identify an Entity from UM-Sent Data and Metadata. 2. MM-Embed the Entity at an M-Location with a Spatial Attitude.		
Request-Action	Time	Request is issued	
	Source	UserID	
	Destination	DeviceID	
	Action	*UM-Embed*	
	InItem	scene ∧ Spatial Attitude	
	InLocation	U-LocationID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Entity ∨ EntityID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Clash	Item clashes with another Item
		M-Location	Out of range
		U-Location	Out of range

8.5.4 UM-Send

Purpose	The Action of transmitting Data & Metadata from the Universe to a Process.		
Request-Action	Time	Request is issued	
	Source	DeviceID	
	Destination	ProcessID	
	Action	*UM-Send*	
	InItem	DataMdata	
	InLocation	DeviceID ∨ Address	
	OutLocation	ProcessID	
	OutRights	Metadata	
Response-Action	Success	OutItem	DataMdata
	Error	Request	Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		U-Location	Out of range

9 Scripting Language

The MPAI-MMM Scripting Language – MMM-Script in the following – serves the double purpose of providing:

1. A handy tool to describe the Actions performed by Processes in an M-Instance.
2. A compact form that a Process can use to request another Process to perform Actions.

9.1 MMM-Script for Action Description

The performance of any Action in an M-Instance is expressed as:

Process ActsOn Item | DataMdata | Media

At	Service | User | MLoc | ULoc	Where Item ends up being located with SA
By	Service	Used to perform *Action*
From	Address | ULoc	Address, ULoc where the source is located
Into	Item	Action leads to
Of	User	Item refers to
To	Address | Device | Process | User	Item/Process where Items ends up being placed
With	DataMdata | Stream	Additional Item required to perform Action

Note: SA is used as a compact form for Spatial Attitude.

Table 5 lists the possible combinations of Actions. Composite Actions are divided into elementary Actions.

Table 5 – Action-Item relationships

Process$_A$	*General Actions*	*Item*	*Indirect object*	*Process$_B$*
human	Registers	Personal Data	*By* M-Instance	Account
User	Changes	Rights	*Of* User	Rights
User	Hides	Item		Rights
User	Identifies	DataMdata	*Into* Item	Item
User	Modifies	Item	*With* DataMdata *Into* Item	Item
Process	Validates	RequestAction	*By* Service	
Process	MM-Sends	RequestAction	*To* Process	
Process	MM-Sends	ResponseAction	*To* Process	
Process	Executes	Contract		

Process$_A$	*Call a Service*			*Process$_B$*
User	Authenticates	Authentication	*At* User	Authentication
User	Authors	DataMdata	*By* Service *At* Service	Item
User	Discovers	DiscoverIn	*By* Service *At* User	Discovery
User	Informs	InformIn	*By* Service *At* User	Information
User	Interprets	InterpretIn	*By* Service *At* User	Interpretation
User	Posts	Item	*At* Service *By* Service	Item
User	Transacts	Item	*To* User *To* Service *By* Service	
User	Converts	Item	*By* Service *Into* Item *At* User	Item
	Resolves (compos.)	Item	By Service Into Item At User	
Process	MM-Sends	ReqAct	*To* Service	ResAct
Service	– MM-Sends	ReqAct	*To* Service	ResAct
Service	– MM-Sends	ReqAct	*To* Process	ResAct
Process$_A$	*Manage Entities (MM)*			*Process$_B$*
User	MM-Adds	Item	*At* M-Location *With* SpAtt	
User	MM-Animates	Model	*At* M-Location *With* Stream *With* SpAtt	
User	MM-Disables	Item	*At* M-Location	
	MM-Embeds (compos.)			
User	– MM-Adds	Item	*At* M-Location *With* SpAtt	
User	– MM-Enable	Item	*At* M-Location	

User	MM-Enables	Item	*At* M-Location	
Process	MM-Sends	Item, DataMdata	*To* Process	
Process$_A$	*Manage Entities (MU)*			
Device	MU-Actuates	Item	*At* U-Location *With* SpAtt	
	MU-Renders (compos.)			
User	– MM-Sends	Item	*To* Device	
Device	– MU-Actuates	Item	*At* U-Location *With* SpAtt	
Process	MU-Sends	Item	*To* Address	
	Track (compos.)			
User	– MM-Embeds	Model	*At* U-Location, *With* SpAtt	
User	– UM-Animates	Model	*With* Stream *At* M-Location	
User	– MU-Renders	Item	*At* U-Location, *With* SpAtt	
Process$_A$	*Manage Entities (UM)*			
	UM-Animates (compos.)			
Device	– UM-Captures	Stream	*From* U-Location	
Device	– UM-Sends	Stream	*To* User	
User	– Identifies	Stream	*Into* Item	
User	– MM-Animates	Model	*With Stream At M-Location At* U-Location, *With* SpAtt	

Device	UM-Captures	Media	*From* U-Location *To* Device	
	UM-Renders (compos.)			
Device	– UM-Captures	Media	*From* U-Location	
Device	– UM-Sends	DataMdata	*To* User	
User	– Identifies	DataMdata	*Into* Item	
User	– MM-Embeds	Item	*At* M-Location *With* SpAtt	
Device	UM-Sends	DataMdata	*From* Address *To* Process	

9.2 Definition in Backus-Naur form

program :=

| /* empty */
| one_or_more_statements
one_or_more_statements :=

| statement
| statement one_or_more_statements
statement :=

| id action_keyword id modifiers
action_keyword :=

|“Register”
| “Change”
| “Hide”
| “Authenticate”
| “Identify”
| “Modify
| “Validate”
| “Execute”
| “Author”
| “Discover”
| “Inform”
| “Interpret”
| “Post”
| “Transact”
| “Convert”
| “Resolve”
| “MM-Add”
| “MM-Animate”
| “MM-Disable”

| “MM-Embed”
| “MM-Enable”
| “MM-Send”
| “MU-Actuate”
| “MU-Render”
| “MU-Send”
| “Track”
| “UM-Animate”
| “UM-Capture”
| “UM-Render”
| “UM-Send”

modifiers :=

| /* empty */
| one_or_more_modifiers

one_or_more_modifiers :=

| modifier
| modifier one_or_more_modifiers

modifier :=

| modifier_keyword id

modifier_keyword :=

| “At”
| “By”
| “From”
| “Into”
| “Of”
| “To”
| “With”

id :=

| STRING
| STRING “@” TIME
| URL “@” TIME
| URL “:” STRING “@” TIME